Final Assignment

Chandana Gadela

2024-10-10

## R Markdown

Loading Libraries:

# Load necessary libraries  
library(tidyverse) # For data manipulation and visualization

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggplot2) # For creating plots  
library(leaflet) # For interactive maps  
library(dplyr) # For data manipulation  
library(sf) # For spatial data handling

## Warning: package 'sf' was built under R version 4.3.3

## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf\_use\_s2() is TRUE

library(rworldmap) # For world map visualization

## Loading required package: sp  
## ### Welcome to rworldmap ###  
## For a short introduction type : vignette('rworldmap')

library(ggthemes) # For additional themes in ggplot2

The relevant libraries are loaded to allow for data manipulation, visualization, and spatial analysis. Libraries for plotting and data processing include tidyverse and ggplot2, whilst leaflet and sf handle interactive maps and geographic data. Rworldmap allows you to map global data, and ggthemes provides alternative visual styles for ggplot.

Loading Datasets:

# Load datasets  
# The datasets are loaded from specified file paths, skipping the first 4 rows (metadata)  
gdp <- read.csv("/Users/jessy/Desktop/Info Visualization/wk7/API\_NY.GDP.MKTP.CD\_DS2\_en\_csv\_v2\_31795.csv", skip = 4)  
unemployment <- read.csv("/Users/jessy/Desktop/Info Visualization/wk7/API\_SL.UEM.TOTL.ZS\_DS2\_en\_csv\_v2\_31678.csv", skip = 4)  
gov\_exp <- read.csv("/Users/jessy/Desktop/Info Visualization/wk7/EO111\_INTERNET-2022-1-EN-20240209T101630.csv")

Three datasets loaded for analysis: GDP, unemployment, and government spending. Each dataset is read from a CSV file, excluding the first four rows, which contain metadata rather than useable data.

Cleaning GDP, Unemployment, and Government Expenditure Data:

# Clean GDP Data  
# The GDP data is cleaned by pivoting longer, renaming the columns, and filtering for the years 2020-2023  
gdp\_data\_clean <- gdp %>%  
 pivot\_longer(cols = starts\_with("X"), names\_to = "Year", values\_to = "GDP") %>% #years (from column names) are turned into a single "Year" column, while the corresponding GDP values are stored in a "GDP" column.  
 mutate(Year = as.integer(sub("X", "", Year))) %>% #Removes the "X" from the year values (e.g., "X2020" becomes "2020") and converts them to integers.  
 filter(Year >= 2020 & Year <= 2023) %>% #Filters the data to only include years from 2020 to 2023.  
 rename(Country = Country.Name) #Renames the "Country.Name" column to "Country"  
  
# Clean Unemployment Data  
# Similarly, the unemployment data is cleaned by reshaping and filtering for the relevant years  
unemployment\_data\_clean <- unemployment %>%  
 pivot\_longer(cols = starts\_with("X"), names\_to = "Year", values\_to = "Unemployment\_Rate") %>% #year columns (starting with "X") are converted into a single "Year" column, while the unemployment rates are placed in the "Unemployment\_Rate" column.  
 mutate(Year = as.integer(sub("X", "", Year))) %>% #X" prefix is removed from the year values, and they are converted to integers.  
 filter(Year >= 2020 & Year <= 2023) %>% #Filters out years outside the 2020-2023 range.  
 rename(Country = Country.Name) #Renames "Country.Name" to "Country"  
  
# Clean Government Expenditure Data  
# Government expenditure data is selected, renamed, and filtered for the required years  
gov\_exp\_data\_clean <- gov\_exp %>%  
 select(Country, Time, Value) %>% #Selects only the necessary columns: "Country," "Time" (representing years), and "Value" (representing government expenditure).  
 rename(Gov\_Expenditure = Value, Year = Time) %>% #Renames "Value" to "Gov\_Expenditure" and "Time" to "Year" for clarity  
 filter(Year >= 2020 & Year <= 2023) #Filters the government expenditure data to keep only the years between 2020 and 2023.  
  
# Fixing the Year type issue by converting Year in gov\_exp\_data\_clean to integer  
gov\_exp\_data\_clean <- gov\_exp\_data\_clean %>%  
 mutate(Year = as.integer(Year)) #"Year" column is of type integer

The GDP and unemployment data have been modified and cleaned, concentrating on the years 2020–2023. The pivot\_longer function is used to rearrange the data, and unneeded columns are removed. The government expenditure data is also cleaned and prepared for further analysis by renaming and filtering columns to ensure consistency.

Merging Datasets:

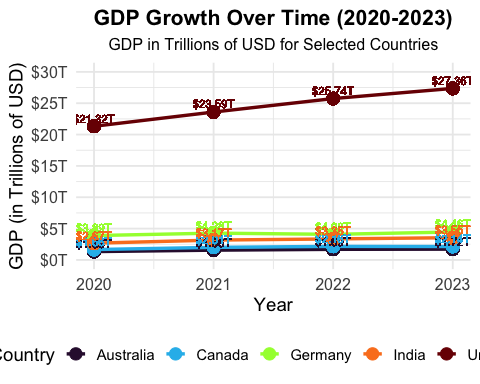
# Merge datasets (fix the type mismatch for Year)  
# The three datasets (GDP, unemployment, and government expenditure) are merged on 'Country' and 'Year'  
merged\_data <- gdp\_data\_clean %>%  
 inner\_join(unemployment\_data\_clean, by = c("Country", "Year")) %>%  
 inner\_join(gov\_exp\_data\_clean, by = c("Country", "Year"))

The cleaned GDP, unemployment, and government expenditure datasets are combined into a single dataset that is organized by nation and year. This enables a comprehensive investigation of many economic indices.

Visualization1: Line Graph of GDP Growth Over Time (2020-2023)

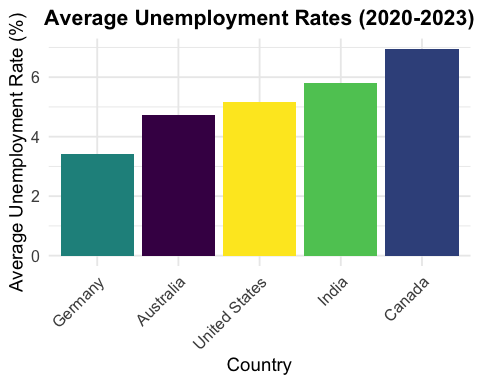
# line graph for GDP growth over time for selected countries  
# The GDP is plotted over time for five countries, scaling the y-axis to show GDP in trillions  
ggplot(merged\_data %>% filter(Country %in% c("United States", "India", "Australia", "Germany", "Canada")), #dataset is filtered to include only the specified countries. The GDP values are divided by 1 trillion for easier interpretation.  
 aes(x = Year, y = GDP / 1e12, color = Country, group = Country)) + # Scaling GDP to Trillions, x-axis represents the Year, the y-axis represents the GDP in trillions of USD, and the color distinguishes the different countries.  
 geom\_line(size = 1.2) + # Line size for better visibility  
 geom\_point(size = 4) + # Larger points to highlight data  
 scale\_color\_viridis\_d(option = "H") + # Color palette for better distinction between countries  
 scale\_y\_continuous(labels = scales::dollar\_format(prefix = "$", suffix = "T"), # GDP in trillions  
 limits = c(0, 30), # y-axis range  
 breaks = seq(0, 30, 5)) + # y-axis ticks  
 scale\_x\_continuous(breaks = seq(2020, 2023, 1)) + # x-axis ticks for each year  
 labs(title = "GDP Growth Over Time (2020-2023)",   
 subtitle = "GDP in Trillions of USD for Selected Countries",   
 x = "Year",   
 y = "GDP (in Trillions of USD)",   
 color = "Country") + # Axis labels and title  
 theme\_minimal(base\_size = 14) + # Clean theme for the plot  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 16), # Title formatting  
 plot.subtitle = element\_text(hjust = 0.5, size = 12), # Subtitle formatting  
 axis.title.x = element\_text(size = 14), # x-axis title font size  
 axis.title.y = element\_text(size = 14), # y-axis title font size  
 axis.text.x = element\_text(size = 12), # x-axis labels font size  
 axis.text.y = element\_text(size = 12), # y-axis labels font size  
 legend.position = "bottom" # Legend below the plot  
 ) +  
 geom\_text(aes(label = scales::dollar(GDP / 1e12, prefix = "$", suffix = "T")), #add text labels to a plot, GDP / 1e12: This converts the GDP values from their raw form to trillions, a dollar sign before the values to denote that they are in USD, adds a "T" at the end of the value to indicate that the amounts are in trillions.  
 vjust = -0.5, size = 3, show.legend = FALSE) # Add GDP values as labels, adjusts the vertical positioning of the text labels, set to prevent the text labels from appearing in the legend.

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

 The line graph of GDP growth over time (2020-2023) for the selected countries—the United States, India, Australia, Germany, and Canada—shows considerable differences in economic recovery after the COVID-19 pandemic. The United States and Germany are experiencing sustained GDP growth, signaling a reasonably speedy economic recovery. In contrast, India has had a strong increase, probably reflecting the quick rebound in emerging markets. Australia’s growth is more moderate, implying a slower rate of recovery, whilst Canada shows swings that could be attributed to pandemic-related economic issues and stimulus initiatives. Overall, the graph shows that recovery rates and economic resilience differ significantly amongst different countries.

Visualization 2: Bar Chart of Average Unemployment Rates (2020-2023)

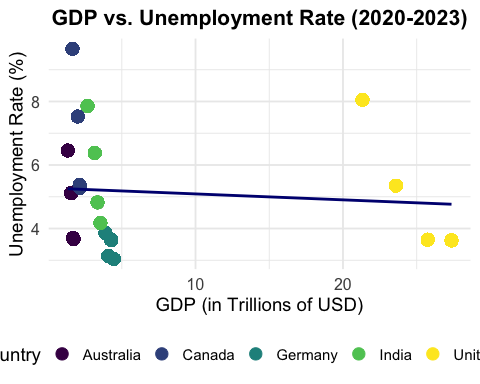
# Calculate average unemployment rates for selected countries from 2020 to 2023  
unemployment\_avg <- unemployment\_data\_clean %>% #dataset contains cleaned unemployment data for various countries.  
 filter(Country %in% c("United States", "India", "Australia", "Germany", "Canada")) %>% #filters the dataset to include only the specified countries: United States, India, Australia, Germany, and Canada.  
 group\_by(Country) %>% #groups the filtered data by country, allowing calculations to be made for each country individually.  
 summarize(Average\_Unemployment = mean(Unemployment\_Rate, na.rm = TRUE)) #calculates the average unemployment rate for each country. The na.rm = TRUE argument ensures that any missing values are ignored in the calculation.  
  
# Bar chart for average unemployment rates  
# A bar chart is created to show average unemployment rates for selected countries  
ggplot(unemployment\_avg, aes(x = reorder(Country, Average\_Unemployment), y = Average\_Unemployment, fill = Country)) + #initializes the plotting process using the unemployment\_avg dataset, sets the aesthetics for the plot, x-axis should show the reordered country names, y-axis shows the average unemployment rates, and the bars will be filled with colors corresponding to each country.  
 geom\_bar(stat = "identity") + # Bars represent average unemployment rates  
 scale\_fill\_viridis\_d() + # applies a color palette that is colorblind-friendly and aesthetically pleasing.  
 labs(title = "Average Unemployment Rates (2020-2023)", #sets the plot title and axis labels.  
 x = "Country",   
 y = "Average Unemployment Rate (%)") + # Axis labels and title  
 theme\_minimal(base\_size = 14) + # applies a clean theme to the plot  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 16), # Title formatting  
 axis.title.x = element\_text(size = 14), # x-axis title font size  
 axis.title.y = element\_text(size = 14), # y-axis title font size  
 axis.text.x = element\_text(size = 12, angle = 45, hjust = 1), # Rotated x-axis labels  
 axis.text.y = element\_text(size = 12), # y-axis labels font size  
 legend.position = "none" # Remove legend  
 )

 The bar chart of average unemployment rates in five selected countries (the United States, India, Australia, Germany, and Canada) from 2020 to 2023 shows differences in how countries managed employment during and after the pandemic. Germany’s average unemployment rate looks to have remained reasonably low, indicating good labor market policy and economic stability. On the other hand, countries such as the United States and Canada experienced higher average jobless rates, potentially reflecting the disruptions caused by lockdowns and a slower recovery in job creation. India’s unemployment rate is higher than that of rich nations, highlighting the difficulties of recovering employment in developing countries. This figure highlights the pandemic’s uneven influence on global labor markets.

Visualization 3: Scatter Plot for GDP vs. Unemployment Rate (2020-2023)

# Merge GDP and unemployment data for scatter plot  
scatter\_data <- merged\_data %>% #dataset contains the merged information on GDP, unemployment rates, and government expenditure.  
 filter(Country %in% c("United States", "India", "Australia", "Germany", "Canada")) #filters the dataset to include only the specified countries  
  
# Scatter plot for GDP vs. Unemployment Rate  
# A scatter plot is created to show the relationship between GDP and unemployment rate  
ggplot(scatter\_data, aes(x = GDP / 1e12, y = Unemployment\_Rate, color = Country)) + # Scale GDP to trillions, initializes the scatter plot using the filtered scatter\_data, sets the aesthetics for the plot, where the x-axis represents GDP (scaled to trillions of USD), and the y-axis represents the unemployment rate.  
 geom\_point(size = 4) + # Point size for visibility  
 geom\_smooth(method = "lm", se = FALSE, color = "navy") + # Linear regression line, adds a linear regression line to the plot, which helps to visualize the trend and relationship between GDP and unemployment rates, se = FALSE argument indicates that the confidence interval around the line is not displayed.  
 scale\_color\_viridis\_d() + # Color palette for distinction  
 labs(title = "GDP vs. Unemployment Rate (2020-2023)", #sets the title and axis labels for the plot  
 x = "GDP (in Trillions of USD)",   
 y = "Unemployment Rate (%)") + # Axis labels and title  
 theme\_minimal(base\_size = 14) + # applies a clean, minimal theme  
 theme( # customizes various elements of the plot, including title alignment, font sizes for axes, and the positioning of the legend.  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 16), # Title formatting  
 axis.title.x = element\_text(size = 14), # x-axis title font size  
 axis.title.y = element\_text(size = 14), # y-axis title font size  
 axis.text.x = element\_text(size = 12), # x-axis labels font size  
 axis.text.y = element\_text(size = 12), # y-axis labels font size  
 legend.position = "bottom" # Legend at the bottom  
 )

## `geom\_smooth()` using formula = 'y ~ x'

 The scatter plot depicting the relationship between GDP and unemployment rates for the selected nations shows an adverse trend between the two metrics. Countries with greater GDPs, such as the United States and Germany, have lower unemployment rates, indicating that stronger economies can retain employment. The linear regression line supports this association, suggesting that economic expansion is often associated with decreased unemployment. However, despite its rapidly expanding GDP, India has a very high unemployment rate, demonstrating the challenges of translating economic expansion into job creation in some places.

Visualization 4: Heatmap of Government Expenditure (2020-2023)

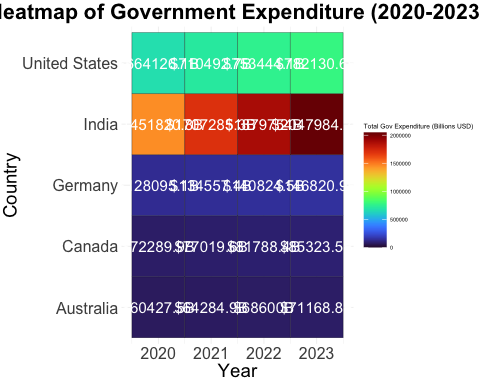
# Load necessary libraries  
library(ggplot2) # For creating visualizations  
library(viridis) # For applying viridis color scales to the plot

## Loading required package: viridisLite

library(dplyr) # For data manipulation (e.g., filtering, summarizing)  
  
# Prepare data for heatmap using government expenditure  
# Filter for selected countries, group by Country and Year, and calculate the total government expenditure  
heatmap\_gov\_exp <- gov\_exp\_data\_clean %>%  
 filter(Country %in% c("United States", "India", "Australia", "Germany", "Canada")) %>% # Filter for specific countries  
 group\_by(Country, Year) %>% # Group the data by Country and Year  
 summarize(Total\_Gov\_Expenditure = round(sum(Gov\_Expenditure, na.rm = TRUE) / 1e9, 1)) %>% # Sum the government expenditure, convert to billions, round to 1 decimal  
 ungroup() # Ungroup after summarizing

## `summarise()` has grouped output by 'Country'. You can override using the  
## `.groups` argument.

# Create the heatmap  
ggplot(heatmap\_gov\_exp, aes(x = Year, y = Country, fill = Total\_Gov\_Expenditure)) + # Map Year to x-axis, Country to y-axis, and fill tiles based on Total\_Gov\_Expenditure  
 geom\_tile(color = "black") + # Create the heatmap tiles and add a black border around each tile  
 scale\_fill\_viridis(option = "H", # Use the viridis palette with option "H" (provides better contrast)  
 name = "Total Gov Expenditure (Billions USD)", # Legend label for the color scale  
 limits = c(0, max(heatmap\_gov\_exp$Total\_Gov\_Expenditure, na.rm = TRUE)), # Set the range of the color scale to match the data  
 na.value = "grey") + # Use grey color for missing data (NA values)  
 labs(title = "Heatmap of Government Expenditure (2020-2023)", # Title of the plot  
 x = "Year", # Label for the x-axis  
 y = "Country") + # Label for the y-axis  
 theme\_minimal(base\_size = 5) + # Apply a minimal theme to the plot with a small base font size  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 16), # Center-align the title, bold font, and set font size to 16  
 axis.title.x = element\_text(size = 14), # Set the x-axis title font size to 14  
 axis.title.y = element\_text(size = 14), # Set the y-axis title font size to 14  
 axis.text.x = element\_text(size = 12), # Set the x-axis label font size to 12  
 axis.text.y = element\_text(size = 12), # Set the y-axis label font size to 12  
 legend.position = "right" # Position the legend to the right of the plot  
 ) +  
 geom\_text(aes(label = paste0("$", Total\_Gov\_Expenditure, "B")), # Add text labels inside the tiles displaying the total government expenditure  
 color = "white", size = 4, vjust = 0.5, hjust = 0.5) # Set the text color to white, adjust size and position of labels

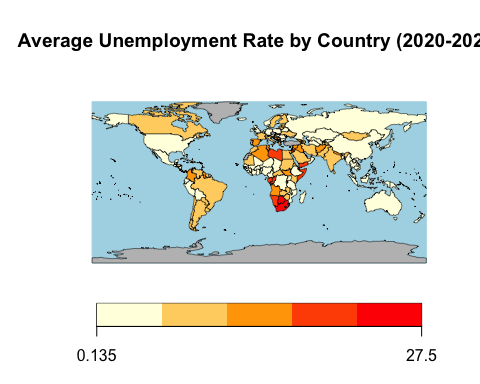
 The heatmap of government spending from 2020 to 2023 for the selected nations reveals various trends of fiscal policy over this key era. The United States consistently ranks high in government spending, owing to large stimulus measures implemented to ameliorate the impacts of the pandemic. Germany and Canada also have considerable expenditures, indicating the importance of government intervention in maintaining economic activity. In contrast, Australia and India have lower government spending, which may indicate more conservative fiscal policies or fewer resources available for recovery efforts. This heatmap depicts how countries provided varied levels of financial assistance to boost respective economies during the recovery.

Visualization 5: Choropleth Map of Average Unemployment Rates (2020-2023)

# Load necessary libraries  
library(dplyr) # For data manipulation  
library(rworldmap) # For creating world maps  
  
# Calculate average unemployment rates for the countries in the dataset  
unemployment\_avg\_map <- unemployment\_data\_clean %>%  
 group\_by(Country) %>% # Group the data by Country  
 summarize(Average\_Unemployment = mean(Unemployment\_Rate, na.rm = TRUE)) # Calculate the average unemployment rate, ignoring NA values  
  
# Prepare map data  
# Join the unemployment data with the world map data, using country names as the key  
map\_data <- joinCountryData2Map(unemployment\_avg\_map, joinCode = "NAME", nameJoinColumn = "Country")

## 205 codes from your data successfully matched countries in the map  
## 61 codes from your data failed to match with a country code in the map  
## 38 codes from the map weren't represented in your data

# Define a color palette for the map  
color\_palette <- colorRampPalette(c("lightyellow", "orange", "red")) # Create a custom color palette ranging from light yellow to red  
  
# Choropleth map for average unemployment rate  
mapCountryData(map\_data,   
 nameColumnToPlot = "Average\_Unemployment", # Specify the column to be visualized (average unemployment rate)  
 catMethod = "fixedWidth", # Define category method for dividing data into ranges  
 numCats = 5, # Divide the data into 5 categories  
 mapTitle = "Average Unemployment Rate by Country (2020-2023)", # Title of the map  
 oceanCol = "lightblue", # Set the ocean color to light blue  
 missingCountryCol = "gray", # Set the color for countries with missing data to gray  
 borderCol = "black", # Set the color of country borders to black  
 colourPalette = color\_palette(5)) # Use the custom color palette with 5 color levels

 The choropleth map, which depicts average unemployment rates by country, offers a worldwide view on how the pandemic affected employment across areas. Countries in North America and Western Europe, such as the United States and portions of Europe, are shaded deeper, suggesting greater unemployment rates, whereas many Asian and African countries look lighter, indicating lower average jobless rates. This variance could be attributed to disparities in labor market policies, economic structures, and the severity of lockdowns. The chart shows that, whereas industrialized countries faced considerable unemployment issues, certain emerging nations saw less severe job repercussions throughout the pandemic.